



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/720,506	11/24/2003	Mitica Manu	MSFT-2792/306045	4588
41505 7590 05/04/2007 WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION) CIRA CENTRE, 12TH FLOOR 2929 ARCH STREET PHILADELPHIA, PA 19104-2891			EXAMINER CHEN, QING	
			ART UNIT 2191	PAPER NUMBER
			MAIL DATE 05/04/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/720,506	Applicant(s) MANU, MITICA	
	Examiner Qing Chen	Art Unit 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 29 March 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-22 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-22 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 29 March 2007 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date <u>20070412</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This Office action is in response to the amendment filed on March 29, 2007.
2. **Claims 1-22** are pending.
3. **Claims 1, 6-9, 12, 13, 15, 17, 19, 20, and 22** have been amended.
4. The objection to the declaration is withdrawn in view of Applicant's arguments.
5. The objections to the drawings are withdrawn in view of Applicant's amendments to the drawings and the specification.
6. The objections to the specification are withdrawn in view of Applicant's amendments to the specification.
7. The objections to Claims 6-8, 12, 13, 15, 17, 19, and 22 are withdrawn in view of Applicant's amendments to the claims. However, Applicant's amendments to the claims fail to fully address the objection to Claims 16 and 20 due to a typographical error. Accordingly, this objection is maintained and further explained below.
8. The 35 U.S.C. § 112, second paragraph, rejections of Claims 7 and 9 are withdrawn in view of Applicant's amendments to the claims.
9. The 35 U.S.C. § 101 rejection of Claim 22 is withdrawn in view of Applicant's amendments to the claim. However, the 35 U.S.C. § 101 rejections of Claims 1-7 are maintained in view of Applicant's amendments to the claims and further explained below.

Response to Amendment

Claim Objections

10. **Claims 2 and 8-22** are objected to because of the following informalities:

- **Claim 2** contains a typographical error: a period (.) should be added after the limitation body.
- **Claims 8 and 22** recite the limitation “the code elements.” Applicant is advised to change this limitation to read “the plurality of code elements” for the purpose of providing it with proper explicit antecedent basis.
- **Claims 9-21** depend on Claim 8 and, therefore, suffer the same deficiency as Claim 8.
- **Claims 16 and 20** contain a typographical error: a comma (,) before the word “or” should be added to separate the last two elements in the series. Although the use of a comma before “or” in a series is not mandatory, Applicant is advised to make the correction in order to keep the grammatical style consistent throughout the claims.
- **Claim 22** contains a typographical error: a colon (:) should be added after the preamble.

Appropriate correction is required.

Claim Rejections - 35 USC § 101

11. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

12. **Claims 1-7** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 1-7 are directed to systems. However, the recited components of the systems appear to lack the necessary physical components (hardware) to constitute a machine or manufacture under § 101. Although the claims recite the graphical representation as being displayed on a computer, the computer is interpreted as an intended use limitation rather than a hardware component of the system, since the claim language defines the modeler as capable of generating the graphical representation to be displayed on the computer. Therefore, these claim limitations can be reasonably interpreted as computer program modules—software *per se*. Furthermore, the specification discloses that the various techniques may be implemented in connection with software (*see page 11, paragraph [0048]*). Therefore, the claims are directed to systems of functional descriptive material *per se*, and hence non-statutory.

The claims constitute computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical “things.” They are neither computer components nor statutory processes, as they are not “acts” being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program’s functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program’s functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

Claim Rejections - 35 USC § 102

13. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

14. **Claims 1-5, 8-13, and 15-22** are rejected under 35 U.S.C. 102(b) as being anticipated by **Goodwin et al.** (US 6,199,195).

As per **Claim 1**, Goodwin et al. disclose:

- a modeler for defining at least one of a plurality of code elements and a structure of a code block and generating a graphical representation displayed on a computer of the at least one code element and structure of the code block, wherein the modeler processes input comprising a code block of source code from an innermost element to an outermost element and generates from the input a code model comprising a graphical representation of a structure and flow of the code block (*see Figure 1; Figure 3: 302, 304, and 306; Figure 4; Column 8: 6-12, "A developer defines which unified models are passed to each template or sub-template. This can be either a class, package, subclass, or any other unified model-defined object. Other unified models are accessed using control structures. The syntax of the templates supports any number of control structures followed by a block code. Control structures and code can be nested within one another."* and 44-54, "Shown are a number of modeling tools 302, 304, 306 both data modeling 302 and object modeling 304, 306, defining data within a database 308 or defining objects and

Art Unit: 2191

relating these objects to the data within the database 308. These definitions are referred to herein as logical models.”; Column 12: 65-67 through Column 13: 1, “Meta data objects can also be modeled into case tools, such as RATIONAL ROSE. One benefit of using a case tool is the ability to graphically associate meta data objects.”).

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Goodwin et al. further disclose:

- a user interface for receiving the definition of the at least one code element and the structure of the code block (*see Figure 1: 108 and 110; Column 5: 47-49, “... a user interface 108, 110, such as a keyboard 108 and screen 110 on which a graphical user interface (GUI) is implemented.”).*

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Goodwin et al. further disclose:

- a selector for selecting at least one of a plurality of programming languages in which to generate the source code from the functional model (*see Figure 5: 508; Column 13: 42-44, “... the user options for code to be generated are obtained ... ”).*

As per **Claim 4**, the rejection of **Claim 3** is incorporated; and Goodwin et al. further disclose:

- a code generator for receiving the graphical representation of the at least one code element and the structure of the code block and the at least one programming language and

Art Unit: 2191

generating source code in the at least one programming language (*see Figure 3: 330; Column 13: 20-26, "... the code generator 330 is a model driven application that reads the object elements from the schema server 316 and applies a set of known templates 324, 326, 328 for a set of services 320 for a given framework 318 to the object elements of the unified models to produce a set of source code objects 332, 334."*).

As per **Claim 5**, the rejection of **Claim 2** is incorporated; and Goodwin et al. further disclose:

- wherein the at least one programming language comprises C++ (*see Column 13: 52-55, "Thus, the code generator 330 can support the creation of, for example, IDL, JAVA or C++ files ..."*).

As per **Claim 8**, Goodwin et al. disclose:

- processing a block of programming code from an innermost element to an outermost element and generating from the processed block of programming code a functional software model (*see Figure 4; Column 8: 6-12, "A developer defines which unified models are passed to each template or sub-template. This can be either a class, package, subclass, or any other unified model-defined object. Other unified models are accessed using control structures. The syntax of the templates supports any number of control structures followed by a block code. Control structures and code can be nested within one another."* and 53-58, "A repository adaptor tool 312 takes the logical models generated by the modeling tools 302, 304, 306 (logical

Art Unit: 2191

models 202 (FIG. 2)) and the model adaptors 310 as inputs and generates the unified models 206 (FIG. 2) in the unified modeling language.”);

- defining a plurality of code elements within the block of programming code (see Column 8: 44-54, “Shown are a number of modeling tools 302, 304, 306 both data modeling 302 and object modeling 304, 306, defining data within a database 308 or defining objects and relating these objects to the data within the database 308.”);

- specifying a structure of the block of programming code including the plurality of code elements (see Column 8: 44-54, “These definitions are referred to herein as logical models.”); and

- generating from the plurality of code elements and the structure of the block of programming code including the plurality of code elements a graphical representation of the plurality of code elements and flow of the block of programming code (see Column 12: 65-67 through Column 13: 1, “Meta data objects can also be modeled into case tools, such as RATIONAL ROSE. One benefit of using a case tool is the ability to graphically associate meta data objects.”).

As per **Claim 9**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- receiving the definition of the plurality of code elements with the block of programming code and specifying the structure of the block of programming code via a user interface (see Column 8: 53-62, “A repository adaptor tool 312 takes the logical models generated by the modeling tools 302, 304, 306 (logical models 202 (FIG. 2)) and the model

Art Unit: 2191

adaptors 310 as inputs and generates the unified models 206 (FIG. 2) in the unified modeling language. The unified models 206 (FIG. 2) are stored in a schema repository 314 accessed by a schema server 316. The unified models 206 (FIG. 2) from the repository adaptor tool 312 are received by the schema server 316 and stored within the schema repository 314.”).

As per **Claim 10**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- specifying at least one target language in which source code for the graphical representation is to be generated (*see Figure 5: 508; Column 13: 42-44, “... the user options for code to be generated are obtained ...”*).

As per **Claim 11**, the rejection of **Claim 10** is incorporated; and Goodwin et al. further disclose:

- generating the source code in the at least one target language (*see Figure 3: 330; Column 13: 20-26, “... the code generator 330 is a model driven application that reads the object elements from the schema server 316 and applies a set of known templates 324, 326, 328 for a set of services 320 for a given framework 318 to the object elements of the unified models to produce a set of source code objects 332, 334.”*).

As per **Claim 12**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

Art Unit: 2191

- wherein one of the plurality of code elements comprises a variable, comment, constant, object, function, method, prototype, member, data type, callback, delegate, reference, field, variant, property, interface, class, type, enumeration, structure, primitive, array, or event handle (*see Column 8: 44-48, "... defining objects and relating these objects to the data within the database 308."*).

As per **Claim 13**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- wherein one of the plurality of code elements comprises a code relation (*see Column 4: 31-36, "A 'relationship' defines a link between two object classes."*).

As per **Claim 15**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- wherein one of the plurality of code elements comprises an evaluation entity (*see Column 4: 6-8, "An 'object instance' is an embodiment (instantiation) of an object class."*).

As per **Claim 16**, the rejection of **Claim 15** is incorporated; and Goodwin et al. further disclose:

- wherein the evaluation entity comprises one of a method call, a plurality of code entities, a plurality of code relations, or an instantiation of a class (*see Column 4: 6-8, "An 'object instance' is an embodiment (instantiation) of an object class."*).

As per **Claim 17**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- wherein one of the plurality of code elements comprises a passive entity (*see Column 8: 49-53, "Also shown are a plurality of model adapters 310 for defining a translation of the logical models of the modeling tools 302, 304, 406 into unified models, expressed in a unified modeling language, such as Unified Modeling Language (UML)."*).

As per **Claim 18**, the rejection of **Claim 15** is incorporated; and Goodwin et al. further disclose:

- wherein the passive entity comprises a comment or a modeling diagram (*see Column 8: 49-53, "Also shown are a plurality of model adapters 310 for defining a translation of the logical models of the modeling tools 302, 304, 406 into unified models, expressed in a unified modeling language, such as Unified Modeling Language (UML)."*).

As per **Claim 19**, the rejection of **Claim 8** is incorporated; and Goodwin et al. further disclose:

- wherein one of the plurality of code elements comprises a block entity (*see Column 4: 1-3, "An 'object class' is a set of data (attributes) and functional capabilities (routines) encapsulated into a single logical entity."*).

As per **Claim 20**, the rejection of **Claim 19** is incorporated; and Goodwin et al. further disclose:

- wherein the block entity comprises a method entity, a member entity, a class entity, a namespace entity, or a file entity (*see Column 4: 1-3, "An 'object class' is a set of data (attributes) and functional capabilities (routines) encapsulated into a single logical entity."*).

As per **Claim 21**, the rejection of **Claim 20** is incorporated; and Goodwin et al. further disclose:

- wherein a many-to-many relationship exists between block entities (*see Column 4: 31-36, "A 'relationship' defines a link between two object classes." and "Relationships can be one-to-one, one-to-many, or many-to-many."*).

Claim 22 is a computer-readable storage medium claim corresponding to the method claim above (Claim 8) and, therefore, is rejected for the same reason set forth in the rejection of Claim 8.

Claim Rejections - 35 USC § 103

15. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

16. **Claims 6 and 7** are rejected under 35 U.S.C. 103(a) as being unpatentable over Goodwin et al. (US 6,199,195).

As per **Claim 6**, the rejection of **Claim 2** is incorporated; however, Goodwin et al. do not disclose:

- wherein the at least one programming language comprises C#.

Official Notice is taken that it is old and well known within the computing art to include C# as one of the programming languages. C# is an object-oriented programming language that is used widely for developing software components suitable for deployment in distributed environments. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include wherein the at least one programming language comprises C#. The modification would be obvious because one of ordinary skill in the art would be motivated to provide support for the most commonly used programming languages.

As per **Claim 7**, the rejection of **Claim 2** is incorporated; however, Goodwin et al. do not disclose:

- wherein the at least one programming language comprises VB Script.

Official Notice is taken that it is old and well known within the computing art to include VB Script as one of the programming languages. VB Script has been used extensively by software programmers to develop a wide variety of applications. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include wherein the at least one programming language comprises VB Script. The modification would be obvious because one of ordinary skill in the art would be motivated to provide support for the most commonly used programming languages.

17. **Claim 14** is rejected under 35 U.S.C. 103(a) as being unpatentable over Goodwin et al. (US 6,199,195) in view of Bailey et al. (US 6,684,385).

As per **Claim 14**, the rejection of **Claim 13** is incorporated; however, Goodwin et al. do not disclose:

- wherein the code relation comprises a mathematical operator.

Bailey et al. disclose:

- wherein the code relation comprises a mathematical operator (*see Column 8: 30-36, "... each program object typically performs some useful function, such as a Boolean operation (e.g., AND, OR, etc.), a mathematical operation, a data acquisition operation ..., renders some comparison (e.g., less than, greater than, equal to, etc.), and so on. "*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Bailey et al. into the teaching of Goodwin et al. to include wherein the code relation comprises a mathematical operator. The modification would be obvious because one of ordinary skill in the art would be motivated to model all aspects of a software program.

Response to Arguments

18. Applicant's arguments filed on March 29, 2007 have been fully considered, but they are not persuasive.

In the remarks, Applicant argues that:

a) Goodwin is directed to generating source code objects. Modeling tools provide input to a model adaptor that generates a unified model. The unified model and system definition are input to a code generator that generates the source code. Goodwin does not disclose or suggest at least a modeler that receives a block of source code and processes the code from an innermost element to an outermost element to generate a graphical representation of the structure and flow of the code as recited by Applicant's amended independent claims.

Examiner's response:

a) Examiner disagrees. Goodwin et al. clearly disclose a modeler that receives a block of source code and processes the code from an innermost element to an outermost element to generate a graphical representation of the structure and flow of the code (*see Figure 1; Figure 3: 302, 304, and 306; Figure 4; Column 8: 6-12, "A developer defines which unified models are passed to each template or sub-template. This can be either a class, package, subclass, or any other unified model-defined object. Other unified models are accessed using control structures. The syntax of the templates supports any number of control structures followed by a block code. Control structures and code can be nested within one another."* and 44-54, "Shown are a number of modeling tools 302, 304, 306 both data modeling 302 and object modeling 304, 306, defining data within a database 308 or defining objects and relating these objects to the data within the database 308. These definitions are referred to herein as logical models."; Column 12: 65-67 through Column 13: 1, "Meta data objects can also be modeled into case tools, such

Art Unit: 2191

as RATIONAL ROSE. One benefit of using a case tool is the ability to graphically associate meta data objects.").

Note that Applicant did not traverse the Examiner's assertion of Official Notice with regard to Claims 6 and 7. Therefore, the "old and well known within the computing art" statement is taken to be admitted prior art because Applicant has failed to traverse the Examiner's assertion of Official Notice (see MPEP § 2144.03).

Conclusion

19. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The

Art Unit: 2191

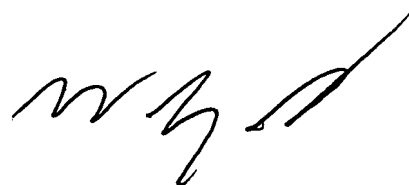
Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM.

The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



WEI ZHEN
SUPERVISORY PATENT EXAMINER

QC / *QC*
April 26, 2007